How Mathematics Can Be Born: A Public Guide to Algebraic Abiogenesis and Mathematical It from Bit

Douglas C. Youvan

doug@youvan.com

www.youvan.ai

July 11, 2026

This article is a nontechnical companion to Douglas C. Youvan's July 2026 paper, *Mathematical It from Bit: Algebraic Abiogenesis, Anonymous Finite Operations, and the Automated Birth of Abstract Algebra*, DOI: 10.13140/RG.2.2.25987.26403. The technical paper asks whether an algebraic theory can be discovered before its objects and axioms have been named. Instead of beginning with groups, rings, lattices, or other familiar structures, Algebraic Abiogenesis begins with small finite sets and anonymous operation tables. A computer forms expressions, evaluates them exactly, and detects expressions that always produce the same result. Repeated equalities become candidate laws. Most candidates are rejected because they are accidental, trivial, redundant, or already known. Surviving laws can define quotient objects, canonical forms, free algebras, model counts, and decision procedures. The achievement so far is not a claim that a major new algebra has already been found. It is the demonstration of a reproducible process by which coherent algebraic theories can emerge from finite operational information. Code, finite tables, and reproducibility records are available at https://github.com/DougYouvan/algebraic-abiogenesis.

Keywords: Algebraic Abiogenesis, Mathematical It from Bit, abstract algebra, artificial intelligence, automated mathematical discovery, finite operations, universal algebra, equation discovery, quotient structures, mathematical objects, freshman mathematics.

A collaboration with OpenAI GPT-5.5 Thinking. CC4.0.

https://orcid.org/0009-0004-4206-4253

1. The Technical Paper and This Public Explanation

The technical paper *Mathematical It from Bit: Algebraic Abiogenesis, Anonymous Finite Operations, and the Automated Birth of Abstract Algebra* develops a formal and computational method for creating candidate algebraic theories from finite anonymous operations.

It includes definitions, algorithms, calibration experiments, formal quotient constructions, term-rewriting arguments, model-count formulas, benchmark proposals, source code, and machine-readable appendices.

The present article explains the same project for readers who have completed high-school mathematics or are beginning college mathematics.

The central question is simple:

Can a mathematical object appear before anyone specifies what kind of object it is supposed to be?

Traditional mathematics usually begins with a definition. Algebraic Abiogenesis begins before the definition.

2. Mathematics Usually Starts After the Important Choices

A first course in abstract algebra may introduce a group as a set G together with an operation satisfying several axioms.

The operation must be associative. There must be an identity element. Every element must have an inverse.

Students then prove theorems from those axioms.

This is normal mathematical practice, but it hides an earlier creative step. Someone had to decide that these particular properties belonged together. Someone had to recognize that many examples shared the same structure.

The same issue appears with rings, fields, vector spaces, lattices, and Boolean algebras. By the time the student encounters them, their operations and laws have already been selected.

Algebraic Abiogenesis asks whether this selection process can itself become a mathematical and computational subject.

Instead of asking only

“What follows from these axioms?”

it asks

“Where could the axioms come from?”

3. Beginning with an Anonymous Operation

Consider the finite set

A = {0,1,2}.

A binary operation takes two members of A and returns one member of A. It can be represented by a table:

*	0	1	2
0	1	2	0
1	2	1	2
2	0	1	0

At the beginning, the symbol * has no interpretation.

It is not called addition.

It is not called multiplication.

It is not assumed to be associative or commutative.

It is simply a finite rule.

A computer can use the table to evaluate expressions such as

x*y,

(x*y)*z,

x*(y*z),

(x*x)*y,

and

x*(y*y).

Before any laws are known, these are different formal expressions.

The computer substitutes every possible value of the variables and records the results.

Suppose it discovers that

(x*x)*y

and

x*y

always produce the same output for every possible x and y.

Then the finite algebra satisfies

(x*x)*y* = *x*y.

The equality is exact. It is not statistical or approximate.

This is the first kind of evidence used by Algebraic Abiogenesis.

4. From Finite Equalities to Candidate Laws

An equation holding in one small table may be accidental.

A three-element system can satisfy an identity that fails immediately on four elements. Several tables may also be closely related because they were generated from the same parent.

The method therefore looks for identities that recur across independently generated operations and, preferably, across different carrier sizes.

Suppose the same equation appears in several unrelated tables of orders three and four.

It then becomes a candidate transferable law.

A collection of such laws may define a class of algebras.

At that point, familiar mathematical questions become possible:

What do all models of the law look like?

Can expressions be reduced to standard forms?

Can equality be decided mechanically?

How many finite models exist?

What are the free algebras?

The development now resembles ordinary abstract algebra, but its starting point was different.

The direction was

anonymous operations
→ exact expression equalities
→ candidate laws
→ algebraic theory.

Traditional algebra usually moves in the reverse direction:

axioms
→ models
→ theorems.

5. Why Generating Equations Is Not Enough

A computer can generate large numbers of equations very easily.

Most are not interesting.

For example, suppose

x*y=x.

The operation ignores its second input. Every complicated expression reduces to whichever variable appears furthest to the left.

This produces many true identities, but the reason is trivial.

A constant operation produces even more equalities because every expression has the same value.

The discovery process must therefore reject operations that are:

constant;

projections;

nearly projections;

dependent on only one coordinate;

or confined to a very small output range.

But even a nondegenerate operation may produce misleading apparent complexity.

Suppose a ternary operation satisfies

m(x,x,x)=x.

Then for every expression t,

m(t,t,t)=t.

Hundreds of different-looking equations may be consequences of this one simple rule.

Counting every consequence as a separate discovery would exaggerate the amount of theory present.

The technical paper therefore introduces explanation-subtracted fertility.

Once a law has been found, the system computes what that law already explains. Those consequences are removed. The search then continues only on the unexplained remainder.

The key question becomes:

What regularity remains after the current explanation has been subtracted?

6. The First Calibration: A Real Theory, but Not a New One

The first complete experiment used one anonymous binary operation.

The search recovered the identities

(x*x)*y*=*x*y,

and

x*(y*y*)=*x*y.

These laws supported substantial mathematical development.

A unary map could be defined by

q(x)=x*x.

The operation was then shown to depend only on q(x) and q(y). This produced a representation theorem, canonical forms, a decidable word problem, and an exact formula for the number of finite labeled models.

At first, the result appeared potentially new.

A later literature audit showed that the theory was already known under an equivalent formulation.

The novelty claim was withdrawn.

This was not a failure of the discovery process. The method had independently recovered authentic mathematics without being given the target theory.

It was, however, a failure of historical classification.

The episode established an important distinction:

A theory may be new to the computer without being new to mathematics.

The technical paper treats this first experiment as a calibration.

7. The Second Calibration: A Ternary Operation

The second complete experiment used one anonymous ternary operation

$m(x,y,z)$.

An early population of finite models appeared to share 41 identities.

The shortest common law was

$m(x,x,x)=x$.

When the consequences of that law were subtracted, all 41 identities disappeared.

What looked like a rich theory was only one shallow mechanism expressed many times.

The search then found eleven models that were individually rich in unexplained identities. However, the eleven did not all support the same deeper theory.

A graph was built.

Each model became a vertex.

Two models were connected when they shared a large residual identity set.

A four-model cluster was found containing three algebras of order three and one of order four.

These four models shared 543 exact identity pairs. Of these, 502 were not explained by diagonal idempotence.

A single stronger law explained the complete common theory:

$m(x,y,y)=x$.

This means that whenever the second and third inputs agree, the operation returns the first input.

The equation gives the reduction rule

$m(u,v,v) \rightarrow u$.

Every use of the rule makes an expression shorter.

This leads to canonical forms: each expression can be simplified until no further reduction applies.

Two expressions are equal in the theory exactly when their canonical forms are the same.

8. Counting the Finite Models

The law

$m(x,y,y)=x$

also makes finite-model counting straightforward.

On a set of n elements, a ternary operation has

n^3

table entries.

The law fixes every entry whose second and third inputs are equal.

There are n choices for x and n choices for y, so the law fixes

n^2

entries.

The remaining

n^3-n^2

entries may be chosen freely.

Each free entry has n possible outputs.

Therefore the number of labeled n-element models is

$N(n)=n^{(n^3-n^2)}$.

For n=2,

$N(2)=2^4=16$.

For n=3,

$N(3)=3^{18}=387,420,489$.

This is an example of how a law discovered from finite tables can become a complete mathematical description of an entire family of algebras.

The equation is not merely a pattern in the original four models. Once adopted as an axiom, it defines infinitely many possible models.

9. How Mathematical Objects Emerge

The phrase Mathematical It from Bit refers to the formation of mathematical objects from finite informational distinctions.

Initially, every formal expression is regarded as different.

For example,

m(x,y,y)

and

x

are different syntax trees.

When a theory establishes

m(x,y,y)=x,

the two expressions become interchangeable.

The same process applies to many other expressions.

This interchangeability creates equivalence classes.

An equivalence class is a collection of different expressions treated as presentations of one mathematical object.

The collection of all such classes forms a quotient algebra.

This kind of construction is already common in mathematics.

Fractions such as

1/2,

2/4,

and

50/100

are different written expressions but represent the same rational number.

Integers that differ by a multiple of n represent the same residue class modulo n.

Words in generators may represent the same group element after defining relations are imposed.

Algebraic Abiogenesis extends this familiar idea by attempting to discover the equivalence policy from operational behavior instead of specifying it entirely in advance.

The progression is

finite information
→ repeated equality
→ stable equivalence
→ quotient
→ mathematical object.

10. What the Artificial Intelligence Does

The computational system is useful because it can perform large exact searches.

It can:

generate finite operation tables;

construct formal expressions;

evaluate every expression on every assignment;

find exact term-function collisions;

compare theories across models;

subtract explained identities;

construct residual-overlap graphs;

and verify finite counts.

Human mathematical reasoning remains necessary.

A person must still judge whether a result is important, identify structural interpretations, prove general theorems, compare the theory with prior literature, and decide what claims are justified.

Language models may help formulate proofs, representations, or explanations, but they can also produce plausible errors.

The present system is therefore hybrid.

The finite searches and exact evaluations are computational.

The interpretation, proof development, novelty assessment, and scientific framing involve human supervision.

The technical paper and its GitHub repository attempt to record this division clearly.

11. Why the GitHub Repository Matters

The technical paper is accompanied by a public reproducibility repository:

https://github.com/DougYouvan/algebraic-abiogenesis

The repository contains:

Python source code;

finite ternary operation tables;

exact identity records;

overlap matrices;

the theory birth-certificate schema;

methodology documentation;

tests;

checksums;

and a one-command reproduction script.

The archived ternary experiment can be rerun to reproduce the principal counts:

300 generated terms;

41 initial common identities;

zero unexplained identities after diagonal-idempotence subtraction;

eleven locally fertile models;

a four-model cross-order clique;

543 common identity pairs;

502 identities unexplained by diagonal idempotence;

and zero unexplained identities after applying m(x,y,y)=x.

The repository allows readers to inspect the finite evidence rather than relying only on the prose description.

It also separates four different questions:

Did the computer produce the reported finite results?

Are the general mathematical arguments correct?

Was the theory already known?

Is the theory important?

Code can strongly address the first question.

Proof addresses the second.

Literature review addresses the third.

Mathematical judgment addresses the fourth.

12. What Has Been Demonstrated

The project has demonstrated that an operation-first process can begin with anonymous finite tables and end with a coherent algebraic theory.

The process has been completed in two different signatures:

one binary operation;

and one ternary operation.

It has produced:

candidate laws;

quotient interpretations;

canonical normal forms;

decidable equality;

free-algebra descriptions;

and exact finite-model counts.

It has also rejected several apparent discoveries.

The binary theory was reclassified as known.

The first ternary identity family was reduced to one shallow law.

Several individually fertile models were rejected because they did not share one common theory.

These rejections are part of the achievement.

An automated discovery process should not be judged only by how many results it produces. It should also be judged by how effectively it eliminates false or exaggerated results.

13. What Has Not Yet Been Demonstrated

The project has not yet produced a theory that has been established as both historically new and mathematically important.

It has not yet shown that operation-first search outperforms direct enumeration of equations.

It has not yet completed a large preregistered benchmark across many signatures and random seeds.

It has not automated every part of the process.

In particular, structural interpretation, formal proof, and literature-equivalence checking still require substantial assistance.

These limitations define the next research stages.

A strong future result would compare the complete method with:

random operation generation;

collision-only evolution;

fixed-operation equation mining;

and direct law enumeration.

A still stronger result would be a candidate theory that survives:

cross-order replication;

explanation subtraction;

adversarial countermodels;

formal verification;

and specialist literature review.

14. Why This Matters to Beginning Mathematics Students

Beginning students often experience mathematics as a finished structure.

Definitions appear in textbooks as though they were inevitable.

Theorems follow.

Exercises test understanding.

Algebraic Abiogenesis reveals another side of mathematics.

Definitions themselves can be discoveries.

The laws chosen to define an object are compressed descriptions of repeated structural behavior.

A mathematical object may be understood not as a mysterious thing behind its formulas, but as a stable system of allowed substitutions and identifications.

This perspective connects freshman mathematics with deeper ideas.

Simplifying fractions is quotient formation.

Reducing algebraic expressions is normal-form computation.

Checking an identity on a finite table is model evaluation.

Finding a counterexample is theory refinement.

Selecting a small set of laws is compression.

Abstract algebra is therefore not only the study of strange symbols. It is the study of which distinctions matter and which can be removed without changing structure.

15. Conclusion

The technical paper *Mathematical It from Bit: Algebraic Abiogenesis, Anonymous Finite Operations, and the Automated Birth of Abstract Algebra* proposes that the formation of algebraic objects can itself be studied experimentally.

The method begins with anonymous finite operations.

It constructs expressions.

It detects exact equivalences.

It rejects trivial or accidental regularities.

It compresses surviving identities into laws.

It forms quotient objects.

It develops those objects into algebraic theories.

The present results are calibrations rather than a claim of a revolutionary new algebra.

Their importance lies in demonstrating a reproducible theory-birth process.

Most mathematics asks:

What follows from this definition?

Algebraic Abiogenesis asks an earlier question:

How can a definition become visible before anyone knows what object they are looking for?

That question connects finite computation, artificial intelligence, universal algebra, and the philosophy of mathematical objects.

The process can be summarized as:

information becomes invariance;

invariance becomes equivalence;

equivalence becomes quotient;

quotient becomes object;

object becomes theory.

16. References

1. Birkhoff, G. “On the Structure of Abstract Algebras.” Proceedings of the Cambridge Philosophical Society, vol. 31, 1935, pp. 433–454.
2. Burris, S., and H. P. Sankappanavar. A Course in Universal Algebra. Springer, 1981.
3. Baader, F., and T. Nipkow. Term Rewriting and All That. Cambridge University Press, 1998.
4. Colton, S. Automated Theory Formation in Pure Mathematics. Springer, 2002.
5. McCune, W. “Solution of the Robbins Problem.” Journal of Automated Reasoning, vol. 19, 1997, pp. 263–276.
6. Schmidt, M., and H. Lipson. “Distilling Free-Form Natural Laws from Experimental Data.” Science, vol. 324, no. 5923, 2009, pp. 81–85.
7. Youvan, D. C. *Mathematical It from Bit: Algebraic Abiogenesis, Anonymous Finite Operations, and the Automated Birth of Abstract Algebra*. July 2026. DOI: 10.13140/RG.2.2.25987.26403.
8. Youvan, D. C. *Algebraic Abiogenesis Methodology and Reproducibility Repository*. GitHub, 2026. https://github.com/DougYouvan/algebraic-abiogenesis.

The author has published over 5,700 AI-assisted papers at: https://www.researchgate.net/profile/Douglas-Youvan/research . Google Scholar has indexed these preprints, and if you want a particular reference, the AI mode of a Google search (Gemini) has efficiently catalogued these preprints.